

An adaptive hybrid stress transition quadrilateral finite element method for linear elasticity

Feiteng Huang ^{*} Xiaoping Xie [†] Chen-Song Zhang [‡]

Abstract

In this paper, we discuss an adaptive hybrid stress finite element method on quadrilateral meshes for linear elasticity problems. To deal with hanging nodes arising in the adaptive mesh refinement, we propose new transition types of hybrid stress quadrilateral elements with 5 to 7 nodes. In particular, we derive a priori error estimation for the 5-node transition hybrid stress element to show that it is free from Poisson-locking, in the sense that the error bound in the a priori estimate is independent of the Lamé constant λ . We introduce, for quadrilateral meshes, refinement/coarsening algorithms, which do not require storing the refinement tree explicitly, and give an adaptive algorithm. Finally we provide some numerical results.

Keywords. Hybrid stress element, transition element, adaptive method, quadrilateral mesh, Poisson-locking, plane elasticity

1 Introduction

Let $\Omega \subset \mathbb{R}^2$ be a convex polygonal domain, with boundary $\Gamma = \Gamma_N \cup \Gamma_D$ and $\text{meas}(\Gamma_D) > 0$. Let \mathbf{n} be the outward unit normal vector on Γ . The plane linear elasticity problem reads

$$\begin{cases} -\mathbf{div} \sigma = \mathbf{f} & \text{in } \Omega \\ \sigma = \mathbb{C}\varepsilon(\mathbf{u}) & \text{in } \Omega \\ \sigma \cdot \mathbf{n}|_{\Gamma_N} = \mathbf{g}, \quad \mathbf{u}|_{\Gamma_D} = 0 \end{cases} \quad (1.1)$$

where $\sigma \in \mathbb{R}_{\text{sym}}^{2 \times 2}$ is the symmetric stress tensor, $\mathbf{u} \in \mathbb{R}^2$ the displacement field, $\varepsilon(\mathbf{u}) = \frac{1}{2}(\nabla + \nabla^T)\mathbf{u}$ the strain tensor, $\mathbf{f} \in \mathbb{R}^2$ the body loading density, and $\mathbf{g} \in \mathbb{R}^2$ the surface traction. Here \mathbb{C} denotes the elasticity modulus tensor with $\mathbb{C}\varepsilon(\mathbf{u}) = 2\mu\varepsilon(\mathbf{u}) + \lambda\text{div}(\mathbf{u})\mathbb{I}$ and \mathbb{I} is the 2×2 identity tensor. The constants μ, λ are the Lamé parameters, given by $\mu = \frac{E}{2(1+\nu)}$, $\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)}$ for plane strain problems and by $\mu = \frac{E}{2(1+\nu)}$, $\lambda = \frac{E\nu}{(1+\nu)(1-\nu)}$ for plane stress problems, where $0 < \nu < 0.5$ is the Poisson's ratio and E is the Young's modulus.

^{*}School of Mathematics, Sichuan University, Chengdu, 610064, China. Email: hftenger@gmail.com

[†]School of Mathematics, Sichuan University, Chengdu, 610064, China. Corresponding author. Email: xpx-iec@gmail.com

[‡]Academy of Mathematics and System Sciences, Beijing, 100190, China. Email: zhangcs@lsec.cc.ac.cn

Hybrid stress finite element method (also called assumed stress hybrid finite element method), based on Hellinger–Reissner variational principle and pioneered by Pian [22], is known to be an efficient approach [24, 23, 25, 32, 33, 35] to improve the performance of the standard 4-node compatible displacement quadrilateral (bilinear) element, which yields poor results for problems with bending and, for plane strain problems, at the nearly incompressible limit. In [24] Pian and Sumihara derived a robust 4-node hybrid stress quadrilateral element (abbr. PS) through a rational choice of stress terms. Xie and Zhou [32, 33] proposed accurate 4-node hybrid stress quadrilateral elements by optimizing stress modes with a so-called energy-compatibility condition [38]. Yu, Xie and Carstensen [35] analyzed the methods and obtained uniform convergence and a posteriori error estimation [24, 32]. It is worth noticing that the 4-node hybrid stress finite element method is of almost the same computational cost as the bilinear Q4 element due to the local elimination of stress parameters.

Adaptive mesh refinement (AMR) for the numerical solution of the PDEs is a standard tool in science and engineering to achieve better accuracy with minimum degrees of freedom. The typical structure in one iteration of adaptive algorithms consists of four steps:

Solve \longrightarrow Estimate \longrightarrow Mark \longrightarrow Refine/Coarsen.

AMR methods locally refine/coarsen meshes according to the estimated error distribution through repeating the above working loop comprised of finite element solution, error estimation, element (edge or patch) marking, and mesh refinement/coarsening until the error decreases to a prescribed level. Classical recursive bisection and coarsening algorithms [26, 27, 18] are widely used in adaptive algorithms (see, for example, ALBERTA [28] and deal.II [2]). These algorithms make use of a refinement tree data structure and subroutines to store/access the refinement history.

Chen and Zhang [9] proposed a non recursive refinement/coarsening algorithm for triangular meshes which does not require storing the bisection tree explicitly. They only store coordinates of vertices and connectivity of triangles which are the minimal information required to represent a mesh for standard finite element computation. In fact, they build the bisection tree structure implicitly into a special ordering of the triangles and simplify the implementation of adaptive mesh refinement and coarsening—thus provided an easy-access interface for the usage of mesh adaptation without much sacrifice in computing time. These algorithms have been extended to 3D later by Bartels and Schreier [3].

Refinement and coarsening for adaptive quadrilateral meshes are more difficult than the counterparts for triangular meshes. When a 4-node quadrilateral element is subdivided into four smaller elements, hanging nodes might appear on the element boundaries of its immediate neighborhoods. There are several different approaches to deal with the hanging nodes. Borouchaki and Frey [4] presented a method to convert the triangular mesh into a quadrilateral mesh, by which one can use the adaptive triangular mesh generation method and then convert the mesh to a quadrilateral one. Schneiders [29] provided some template elements for

local refinement to connect different layer patterns. This method would keep the conformity of mesh, but at the same time, could introduce distorted elements. Another approach is to introduce transition elements, namely, keep the 'hanging' nodes in the mesh. This kind of mesh is called 1-irregular mesh, which is widely used in the field of adaptive quadrilateral finite element methods.

Gupta [16] derived a set of compatible interpolation functions for the quadrilateral transition elements. The displacement interpolation along a 3-node edge is continuous piecewise bilinear instead of quadratic, thus preserves the inter-element compatibility. McDill [20] and Morton [21] extended Gupta's conforming transition elements to 3D. Choi et al. [12, 11, 13, 10] proposed a set of 2D and 3D nonconforming transition elements. Carstensen and Hu [6] provided a method to preserve the inter-element compatibility with just modifying the nodal bases of the immediate neighborhoods of the hanging nodes. In [17] Huang and Xie proved that the consistency error of Choi and Park's 5-node nonconforming transition quadrilateral element [12, 13] is of only $O(h^{1/2})$ -accuracy on transition edges of the quadrilateral subdivision. By modifying the shape functions with respect to edge mid-nodes, the authors obtained a transition element with improved consistency error of order $O(h)$. Zhao, Shi, and Du [37] further extended the element to higher orders and established a posteriori error reliability and efficiency analysis.

For the plane elasticity problem (1.1), Lo, Wan, and Sze developed 4-node to 7-node hybrid stress transition elements, using Gupta's conforming displacement interpolation functions [16] and corresponding 5-parameter to 11-parameter stress modes in skew coordinates. Wu, Sze, and Lo [31] constructed, for 2D and 3D elasticity problems, new enhanced assumed strain (EAS) and hybrid stress transition element families with respect to the incompatible displacement modes of Choi and Park [12, 13].

In this paper, basing on the incompatible displacement interpolation functions by Huang and Xie [17], we propose new 5-node to 7-node hybrid stress transition quadrilateral elements for the elasticity problem (1.1) on adaptive meshes. We derive, for the presented 5-node transition element, a first-order a priori error estimate which is uniform with respect to the Lamé constant λ . Besides, we introduce new refinement/coarsening algorithms for quadrilateral meshes, which are counterparts of the algorithms by Chen and Zhang [9] for triangular meshes. And we present an adaptive finite element method based on the proposed hybrid stress transition elements.

The rest of this paper is organized as follows. In section 2, we present weak formulations for the plane linear elasticity problem. Section 3 shows the construction of new hybrid stress transition elements. Section 4 provides new refinement/coarsening algorithms for quadrilateral meshes and an adaptive hybrid stress finite element method. Finally we give some numerical experiments in Section 5.

2 Weak formulations

We define the following spaces:

$$\mathbf{V} := \left\{ \mathbf{u} \in H^1(\Omega)^2 : \mathbf{u}|_{\Gamma_D} = 0 \right\},$$

$$\Sigma := \begin{cases} \mathbf{L}^2(\Omega; \mathbb{R}_{sym}^{2 \times 2}), & \text{if } meas(\Gamma_N) > 0, \\ \left\{ \tau \in \mathbf{L}^2(\Omega; \mathbb{R}_{sym}^{2 \times 2}) : \int_{\Omega} tr \tau d\Omega = 0 \right\}, & \text{if } \Gamma_N = \emptyset. \end{cases}$$

Here $H^k(T)$ denotes the usual Sobolev space consisting of functions defined on T with derivatives of order up to k being square-integrable, with norm $\|\cdot\|_{k,T}$ and semi-norm $|\cdot|_{k,T}$. In particular, $H^0(T) = L^2(T)$. When there is no conflict, we may abbreviate the norm and semi-norm to $\|\cdot\|_k$ and $|\cdot|_k$, respectively. We use $\mathbf{L}^2(\Omega; \mathbb{R}_{sym}^{2 \times 2})$ to denote the space of square-integrable symmetric tensors with the norm $\|\cdot\|_0$ defined by $\|\tau\|_0^2 := \int_{\Omega} \tau : \tau d\mathbf{x}$, and $tr \tau := \tau_{11} + \tau_{22}$ to represent the trace of τ . We note that on the space \mathbf{V} the semi-norms $|\cdot|_1$, $|\varepsilon(\cdot)|_0$ and the norm $\|\cdot\|_1$ are equivalent due to Korn's inequalities.

Basing on the Hellinger–Reissner variational principle, the weak problem for the model (1.1) reads: Find $(\sigma, \mathbf{u}) \in \Sigma \times \mathbf{V}$, such that

$$a(\sigma, \tau) - b(\tau, \mathbf{u}_h) = 0, \quad \forall \tau \in \Sigma \quad (2.1)$$

$$b(\sigma, \mathbf{v}) = F(\mathbf{v}), \quad \forall \mathbf{v} \in V \quad (2.2)$$

where

$$\begin{aligned} a(\sigma, \tau) &:= \int_{\Omega} \sigma : \mathbb{C}^{-1} \tau d\mathbf{x} = \frac{1}{2\mu} \int_{\Omega} \left(\sigma : \tau - \frac{\lambda}{2(\mu + \lambda)} tr \sigma tr \tau \right) d\mathbf{x}, \\ b(\tau, \mathbf{v}) &:= \int_{\Omega} \tau : \varepsilon(\mathbf{v}) d\mathbf{x}, \\ F(\mathbf{v}) &:= \int_{\Omega} \mathbf{f} \cdot \mathbf{v} d\mathbf{x} + \int_{\Gamma_N} \mathbf{g} \cdot \mathbf{v} ds. \end{aligned}$$

As shown in [35], the following two uniform stability conditions hold for the weak problem (2.1–2.2).

- (A1) Kernel-coercivity: for any $\tau \in Z := \{\tau \in \Sigma : \int_{\Omega} \tau : \varepsilon(\mathbf{v}) d\mathbf{x} = 0, \text{ for all } \mathbf{v} \in \mathbf{V}\}$ it holds $\|\tau\|_0^2 \lesssim a(\tau, \tau)$.
- (A2) Inf-sup condition: for any $\mathbf{v} \in \mathbf{V}$ it holds $|\mathbf{v}|_1 \lesssim \sup_{0 \neq \tau \in \Sigma} \frac{\int_{\Omega} \tau : \varepsilon(\mathbf{v}) d\mathbf{x}}{\|\tau\|_0}$.

Here and in what follows, we use the notation $a \lesssim b$ (or $a \gtrsim b$) [34] to represent that there exists a generic positive constant C , independent of the mesh parameter h and Lamé constant λ , such that $a \leq Cb$ (or $a \geq Cb$). The notation $a \approx b$ abbreviates $a \lesssim b \lesssim a$.

We have the following well-posedness result; see [35].

Proposition 2.1. *Assume that $\mathbf{f} \in L^2(\Omega)^2$, $\mathbf{g} \in H^{1/2}(\Gamma_N)$. Then the weak problem (2.1)–(2.2) admits a unique solution $(\sigma, \mathbf{u}) \in \Sigma \cap H^1(\Omega; \mathbb{R}_{sym}^{2 \times 2}) \times \mathbf{V} \cap H^2(\Omega)^2$ such that*

$$|\sigma|_1 + |\mathbf{u}|_2 \lesssim \|\mathbf{f}\|_0 + \|\mathbf{g}\|_{\frac{1}{2}, \Gamma_N}.$$

3 Hybrid stress transition quadrilateral elements

3.1 Element geometry

Let \mathcal{T}_h be a conventional quadrilateral mesh of Ω . We denote by h_K the diameter of a quadrilateral $K \in \mathcal{T}_h$, and denote $h := \max_{K \in \mathcal{T}_h} h_K$. Let $Z_i(x_i, y_i)$, $1 \leq i \leq 4$ be the four vertices of K , and T_i denotes the sub-triangle of K with vertices Z_{i-1} , Z_i and Z_{i+1} (the index on Z_i is modulo 4).

We assume that the partition \mathcal{T}_h satisfies the following “shape-regularity” hypothesis: there exist a constant $\varrho > 2$ independent of h such that for all $K \in \mathcal{T}_h$,

$$h_K \leq \varrho \rho_K \quad (3.1)$$

with $\rho_K := \min_{1 \leq i \leq 4} \{ \text{diameter of circle inscribed in } T_i \}$.

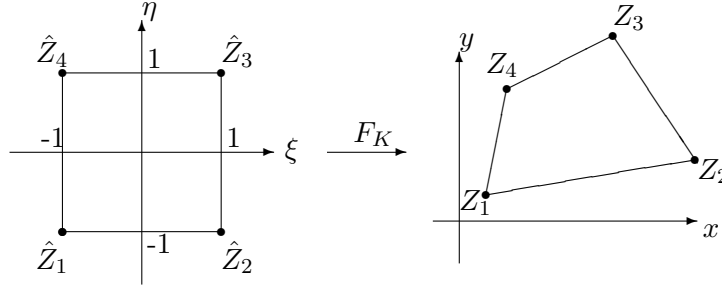


Figure 1: The mapping F_K

We define the bilinear mapping $F_K : \hat{K} = [-1, 1]^2 \longrightarrow K$ (see Figure 1) as

$$\mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix} = F_K(\xi, \eta) = \frac{1}{4} \sum_{i=1}^4 (1 + \xi_i \xi)(1 + \eta_i \eta) \begin{pmatrix} x_i \\ y_i \end{pmatrix}, \quad (3.2)$$

where ξ, η are the local coordinates, and

$$\begin{pmatrix} \xi_1 & \xi_2 & \xi_3 & \xi_4 \\ \eta_1 & \eta_2 & \eta_3 & \eta_4 \end{pmatrix} = \begin{pmatrix} -1 & 1 & 1 & -1 \\ -1 & -1 & 1 & 1 \end{pmatrix}.$$

The Jacobi matrix of the transformation F_K is

$$DF_K(\xi, \eta) = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{pmatrix} = \begin{pmatrix} a_1 + a_{12}\eta & a_2 + a_{12}\xi \\ b_1 + b_{12}\eta & b_2 + b_{12}\xi \end{pmatrix}$$

with

$$\begin{pmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_{12} & b_{12} \end{pmatrix} = \frac{1}{4} \begin{pmatrix} -1 & 1 & 1 & -1 \\ -1 & -1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix} \begin{pmatrix} x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \\ x_4 & y_4 \end{pmatrix}.$$

The Jacobian, J_K , of F_K has the form

$$J_K(\xi, \eta) = \det(DF_K) = J_0 + J_1\xi + J_2\eta$$

with

$$J_0 = a_1b_2 - a_2b_1, \quad J_1 = a_1b_{12} - a_{12}b_1, \quad J_2 = a_{12}b_2 - a_2b_{12}.$$

Under the hypothesis (3.1), it holds the following element geometric properties (see [36]):
For any $K \in \mathcal{T}_h$,

$$\frac{\max_{(\xi, \eta) \in \hat{K}} J_K(\xi, \eta)}{\min_{(\xi, \eta) \in \hat{K}} J_K(\xi, \eta)} < \frac{h_K^2}{2\rho_K^2} \leq \frac{\varrho^2}{2}, \quad (3.3)$$

$$\rho_K^2 < 4(a_1^2 + b_1^2) < h_K^2, \quad \rho_K^2 < 4(a_2^2 + b_2^2) < h_K^2, \quad 4(a_{12}^2 + b_{12}^2) < \frac{1}{4}h_K^2. \quad (3.4)$$

Without loss of generality, we assume

$$|b_1| \leq a_1 \quad \text{and} \quad |a_2| \lesssim b_2. \quad (3.5)$$

Then we have

$$a_1 \approx b_2 \approx h_K, \quad \max\{a_2, b_1\} \lesssim O(h_K), \quad J_K \approx J_0 \approx h_K^2. \quad (3.6)$$

3.2 5-node to 7-node hybrid stress transition elements

Let u_i, v_i ($i = 1, \dots, 8$) be the two components of displacement of the four vertices and four mid-nodes of a transition quadrilateral element K (see Figure 2 for nodal number systems). Following [17], we define the nodal basis N_i ($i = 1, \dots, 8$) as follows:

$$\begin{cases} N_1 = \frac{1}{4}(1 - \xi)(1 - \eta) - \frac{1}{2}(\tilde{N}_7 + \tilde{N}_8), & N_2 = \frac{1}{4}(1 + \xi)(1 - \eta) - \frac{1}{2}(\tilde{N}_8 + \tilde{N}_5), \\ N_3 = \frac{1}{4}(1 + \xi)(1 + \eta) - \frac{1}{2}(\tilde{N}_5 + \tilde{N}_6), & N_4 = \frac{1}{4}(1 - \xi)(1 + \eta) - \frac{1}{2}(\tilde{N}_6 + \tilde{N}_7), \end{cases} \quad (3.7)$$

and

$$N_i = \Delta_i \tilde{N}_i \quad \text{for } i = 5, \dots, 8, \quad (3.8)$$

where

$$\begin{aligned} \Delta_i &= \begin{cases} 1, & \text{if the } i\text{-th node exists (see Figure 2),} \\ 0, & \text{otherwise,} \end{cases} \\ \begin{cases} \tilde{N}_5 = \frac{3}{8}(1 + \xi)(1 - \eta^2), & \tilde{N}_6 = \frac{3}{8}(1 + \eta)(1 - \xi^2), \\ \tilde{N}_7 = \frac{3}{8}(1 - \xi)(1 - \eta^2), & \tilde{N}_8 = \frac{3}{8}(1 - \eta)(1 - \xi^2). \end{cases} \end{aligned} \quad (3.9)$$

The displacement interpolation function \mathbf{v}_{tr} on the transition element K has the form

$$\hat{\mathbf{v}}_{tr} = \mathbf{v}_{tr} \circ F_K = \sum_{i=1}^8 N_i \begin{pmatrix} u_i \\ v_i \end{pmatrix}. \quad (3.10)$$

Remark 3.1. We note that if K is a normal 4-node quadrilateral element, the displacement interpolation \mathbf{v}_{tr} reduces to the standard isoparametric bilinear interpolation \mathbf{v}_{bi} , i.e.

$$\mathbf{v}_{tr} = \hat{\mathbf{v}}_{bi} := \mathbf{v}_{bi} \circ F_K = \sum_{i=1}^4 N_i \begin{pmatrix} u_i \\ v_i \end{pmatrix}. \quad (3.11)$$

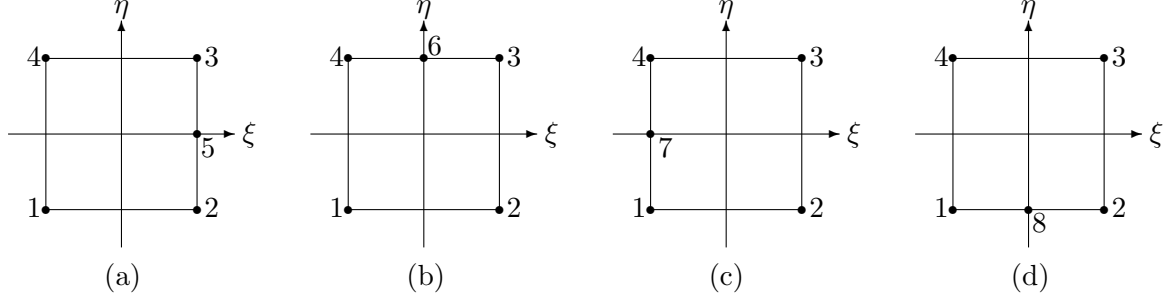


Figure 2: Node number system for transition elements

Let \mathbf{V}_h be a finite dimensional displacement space defined as

$$\mathbf{V}_h := \left\{ \mathbf{v} : \mathbf{v}|_K = \mathbf{v}_{tr} \text{ for } K \in \mathcal{T}_h, \text{ and } \mathbf{v} \text{ vanishes at the nodes on } \Gamma_D \right\}, \quad (3.12)$$

where \mathbf{v}_{tr} is given by (3.10). We define, on \mathbf{V}_h , a semi-norm

$$\|\mathbf{v}\|_h := \left(\sum_{K \in \mathcal{T}_h} \int_K \nabla \mathbf{v} : \nabla \mathbf{v} d\mathbf{x} \right)^{1/2}.$$

It is easy to see $\|\cdot\|_h$ is also a norm on \mathbf{V}_h .

Remark 3.2. From (3.9) and (3.7), it is easy to get the following relation [17]:

$$\int_e [\mathbf{w}] ds = \mathbf{0} \quad \forall \mathbf{w} \in \mathbf{V}_h, \quad \forall e \in \mathcal{E}_h^*, \quad (3.13)$$

where $[\mathbf{w}]$ denotes the jump of function \mathbf{w} across an interior edge e with $[\mathbf{w}] = \mathbf{w}$ when $e \subset \partial\Omega$, and \mathcal{E}_h^* is the set of all 3-node edges of all transition elements in \mathcal{T}_h .

In the following we introduce 5-parameter to 11-parameter stress modes corresponding to arbitrary 4-node to 7-node quadrilateral elements, with parameters $\beta_i \in \mathbb{R}$ for $i = 1, 2, \dots, 11$. We use, for convenience, the Voigt notation $\tau = (\tau_{11}, \tau_{22}, \tau_{12})^T$ to denote a symmetric stress tensor $\tau = \begin{pmatrix} \tau_{11} & \tau_{12} \\ \tau_{12} & \tau_{22} \end{pmatrix}$.

- (1) If K is a 4-node quadrilateral, we use the stress mode of PS [24] or ECQ4 [32] hybrid stress element with $\beta_5^T = (\beta_1, \dots, \beta_5)^T$.

PS stress mode:

$$\hat{\tau}_4 = \begin{pmatrix} 1 & 0 & 0 & \eta & \frac{a_2^2}{b_2^2}\xi \\ 0 & 1 & 0 & \frac{b_1^2}{a_1^2}\eta & \xi \\ 0 & 0 & 1 & \frac{b_1}{a_1}\eta & \frac{a_2}{b_2}\xi \end{pmatrix} \beta_5^\tau \quad (3.14)$$

ECQ4 stress mode:

$$\hat{\tau}_4 = \begin{pmatrix} 1 - \frac{b_{12}}{b_2}\xi & \frac{a_{12}a_2}{b_2^2}\xi & \frac{a_{12}b_2 - a_2b_{12}}{b_2^2}\xi & \eta & \frac{a_2^2}{b_2^2}\xi \\ \frac{b_1b_{12}}{a_1^2}\eta & 1 - \frac{a_{12}}{a_1}\eta & \frac{a_1b_{12} - a_{12}b_1}{a_1^2}\eta & \frac{b_1^2}{a_1^2}\eta & \xi \\ \frac{b_{12}}{a_1}\eta & \frac{a_{12}}{b_2}\xi & 1 - \frac{b_{12}}{b_2}\xi - \frac{a_{12}}{a_1}\eta & \frac{b_1}{a_1}\eta & \frac{a_2}{b_2}\xi \end{pmatrix} \beta_5^\tau. \quad (3.15)$$

(2) If K is a 5-node transition quadrilateral, we use the 7-parameter mode with $\beta_7^\tau = (\beta_1, \dots, \beta_7)^T$:

$$\begin{aligned} \hat{\tau}_5 &= \begin{pmatrix} 1 & 0 & 0 & \eta & 0 & \xi & 0 \\ 0 & 1 & 0 & 0 & \xi & 0 & \eta \\ 0 & 0 & 1 & \frac{b_1^2\xi + b_1b_2\eta}{a_1b_2 - a_2b_1} & \frac{a_1a_2\xi + a_2^2\eta}{a_1b_2 - a_2b_1} & \frac{b_1b_2\xi + b_2^2\eta}{a_2b_1 - a_1b_2} & \frac{a_1^2\xi + a_1a_2\eta}{a_2b_1 - a_1b_2} \end{pmatrix} \beta_7^\tau \\ &=: M_7\beta_7^\tau. \end{aligned} \quad (3.16)$$

(3) If K is a 6-node transition quadrilateral with opposite mid-side nodes, we use the 9-parameter mode with $\beta_9^\tau = (\beta_1, \dots, \beta_7, \beta_8, \beta_9)^T$:

$$\hat{\tau}_6 = M_7\beta_7^\tau + \begin{pmatrix} 2a_2^2\xi\eta - 2a_1a_2\xi^2 & a_2^2\xi^2 \\ 2b_2^2\xi\eta - 2b_1b_2\xi^2 & b_2^2\xi^2 \\ 2a_2b_2\xi\eta - (a_1b_2 + a_2b_1)\xi^2 & a_2b_2\xi^2 \end{pmatrix} \begin{pmatrix} \beta_8 \\ \beta_9 \end{pmatrix}. \quad (3.17)$$

If K is a 6-node transition quadrilateral with adjacent mid-side nodes, we use the 9-parameter mode

$$\hat{\tau}_6 = M_7\beta_7^\tau + \begin{pmatrix} a_1^2\eta^2 & a_2^2\xi^2 \\ b_1^2\eta^2 & b_2^2\xi^2 \\ a_1b_1\eta^2 & a_2b_2\xi^2 \end{pmatrix} \begin{pmatrix} \beta_8 \\ \beta_9 \end{pmatrix} =: M_9\beta_9^\tau. \quad (3.18)$$

(4) If K is a 7-node transition quadrilateral, we use the 11-parameter mode

$$\hat{\tau}_7 = M_9\beta_9^\tau + \begin{pmatrix} 2a_1^2\xi\eta - 2a_1a_2\eta^2 & 2a_2^2\xi\eta - 2a_1a_2\xi^2 \\ 2b_1^2\xi\eta - 2b_1b_2\eta^2 & 2b_2^2\xi\eta - 2b_1b_2\xi^2 \\ 2a_1b_1\xi\eta - (a_1b_2 + a_2b_1)\eta^2 & 2a_2b_2\xi\eta - (a_1b_2 + a_2b_1)\xi^2 \end{pmatrix} \begin{pmatrix} \beta_{10} \\ \beta_{11} \end{pmatrix}. \quad (3.19)$$

Remark 3.3. We now introduce the modified partial derivatives $\frac{\tilde{\partial}}{\partial x}$, $\frac{\tilde{\partial}}{\partial y}$, and corresponding $\tilde{\text{div}}\cdot$, $\tilde{\varepsilon}(\cdot)$ [36]: For any $K \in \mathcal{T}_h$,

$$\begin{aligned} (J_K \frac{\tilde{\partial} v}{\partial x}|_K \circ F_K)(\xi, \eta) &= \frac{\partial y}{\partial \eta}(0, 0) \frac{\partial \hat{v}}{\partial \xi} - \frac{\partial y}{\partial \xi}(0, 0) \frac{\partial \hat{v}}{\partial \eta} = b_2 \frac{\partial \hat{v}}{\partial \xi} - b_1 \frac{\partial \hat{v}}{\partial \eta}, \\ (J_K \frac{\tilde{\partial} v}{\partial y}|_K \circ F_K)(\xi, \eta) &= \frac{\partial x}{\partial \xi}(0, 0) \frac{\partial \hat{v}}{\partial \eta} - \frac{\partial x}{\partial \eta}(0, 0) \frac{\partial \hat{v}}{\partial \xi} = a_1 \frac{\partial \hat{v}}{\partial \eta} - a_2 \frac{\partial \hat{v}}{\partial \xi}, \\ \tilde{\text{div}} \mathbf{v}|_K &= \frac{\tilde{\partial} u}{\partial x} + \frac{\tilde{\partial} v}{\partial y}, \quad \tilde{\varepsilon}(\mathbf{v})|_K = \begin{pmatrix} \frac{\tilde{\partial} u}{\partial x} & \frac{1}{2} \left(\frac{\tilde{\partial} u}{\partial y} + \frac{\tilde{\partial} v}{\partial x} \right) \\ \frac{1}{2} \left(\frac{\tilde{\partial} u}{\partial y} + \frac{\tilde{\partial} v}{\partial x} \right) & \frac{\tilde{\partial} v}{\partial y} \end{pmatrix}. \end{aligned}$$

It is easy to know that the stress modes $\hat{\tau}_i$ defined in (3.16)–(3.19) satisfy the modified equilibrium relation

$$\mathbf{div} \tau := \left(\tilde{div} \begin{pmatrix} \tau_{11} \\ \tau_{12} \end{pmatrix}, \tilde{div} \begin{pmatrix} \tau_{12} \\ \tau_{22} \end{pmatrix} \right)^T = 0 \quad \text{on } K$$

for $\tau|_K = \hat{\tau}_i \circ F_K^{-1}$ and $i = 5, \dots, 7$. In particular, for the 7-parameter stress mode, $\hat{\tau}_5$ in (3.16), of a 5-node transition element, it's easy to verify the relation

$$\int_K \tau : \tilde{\varepsilon}(\mathbf{v}^b) d\mathbf{x} = 0 \quad (3.20)$$

for any $\mathbf{v}^b \in B_h := \{ \mathbf{v} \in L^2(\Omega)^2 : \hat{\mathbf{v}} = \mathbf{v}^b|_K \circ F_K \in \text{span}\{1 - (\xi^2 + \eta^2)/2\}^2, \forall K \in \mathcal{T}_h \}$.

Remark 3.4. We note that the stress modes (3.16)–(3.19) for the 5-node to 7-node transition elements can be viewed as modified versions of those introduced by Lo, Wan and Sze [19]. In particular, these two versions are identical when K is a parallelogram.

Basing on the stress modes (3.14)–(3.19), we define the approximation stress space Σ_h as

$$\Sigma_h = \left\{ \tau \in \Sigma : \hat{\tau} = \tau|_K \circ F_K = \hat{\tau}_i, \text{ if } K \text{ is a } i\text{-node quadrilateral in } \mathcal{T}_h, i = 4, \dots, 7 \right\}.$$

Now we give the hybrid stress finite element scheme for the problem (2.1)–(2.2): find $(\sigma_h, \mathbf{u}_h) \in \Sigma_h \times \mathbf{V}_h$ such that

$$a(\sigma_h, \tau) - b_h(\tau, \mathbf{u}_h) = 0, \quad \forall \tau \in \Sigma_h \quad (3.21)$$

$$b_h(\sigma_h, \mathbf{v}) = F(\mathbf{v}), \quad \forall \mathbf{v} \in \mathbf{V}_h \quad (3.22)$$

where $b_h(\tau, \mathbf{v}) = \sum_{K \in \mathcal{T}_h} \int_K \tau : \varepsilon(\mathbf{v}) d\mathbf{x}$.

3.3 Uniform error estimation for 5-node hybrid stress transition element

To derive uniform error estimates for the hybrid stress method (3.21)–(3.22), we need, according to the mixed finite element method theory [15, 5], the following two discrete versions of the uniform stability conditions (A1) and (A2):

(A1_h) Discrete Kernel-coercivity: For any $\tau \in Z_h := \{ \tau \in \Sigma_h : \sum_K \int_K \tau : \varepsilon(\mathbf{v}) d\mathbf{x} = 0, \forall \mathbf{v} \in \mathbf{V}_h \}$, it holds that $\|\tau\|_0^2 \lesssim a(\tau, \tau)$.

(A2_h) Discrete Inf-sup condition: For any $\mathbf{v} \in \mathbf{V}_h$, it holds that $\|\mathbf{v}\|_h \lesssim \sup_{0 \neq \tau \in \Sigma_h} \frac{\sum_K \int_K \tau : \varepsilon(\mathbf{v}) d\mathbf{x}}{\|\tau\|_0}$.

It has been shown that the uniform stability conditions (A1_h)–(A2_h) hold in the case of 4-node hybrid stress quadrilateral finite element method [35]. In this subsection we will show that they also hold for the proposed 5-node hybrid stress transition element. For the cases of

6-node and 7-node transition elements, one may follow the same method to get similar stability results.

As for $(A1_h)$, following the same procedure as in the proof of Theorem 4.1 in [35] and using Theorem 5.2 of [36] and (3.20), we can easily obtain the following result:

Proposition 3.1. *Let the partition \mathcal{T}_h satisfy the shape-regularity condition (3.1). Assume that for any $\bar{q} \in \bar{W}_h := \{\bar{q} \in L^2(\Omega) : \bar{q}|_K \in P_0(K), \forall K \in \mathcal{T}_h\}$, there exists some $\mathbf{v} \in \mathbf{V}_h$ with*

$$\|\bar{q}\|_0^2 \lesssim \int_{\Omega} \bar{q} \operatorname{div} \mathbf{v} d\mathbf{x}, \quad \|\mathbf{v}\|_h^2 \lesssim \|\bar{q}\|_0^2.$$

Then the uniform discrete Kernel-coercivity condition $(A1_h)$ holds for the 5-node hybrid stress transition element.

Remark 3.5. *The above result implies that any quadrilateral mesh which is stable for the Stokes element $Q1-P0$ satisfies $(A1_h)$. As we know, the only unstable case for $Q1-P0$ is the checkerboard mode. Thereupon, any quadrilateral mesh which breaks the checkerboard mode is sufficient to guarantee the uniform stability condition $(A1_h)$.*

The rest of this subsection is devoted to the proof of the uniform discrete inf-sup condition $(A2_h)$ for the 5-node hybrid stress transition element. Without loss of generality we only consider the cases of (a) and (c) in Figure 2. Thus, from (3.8)–(3.10) we have, for $\mathbf{v} = (u, v)^T \in \mathbf{V}_h$ with nodal values $\mathbf{v}(Z_i) = (u_i, v_i)^T$ on K ,

$$\hat{\mathbf{v}} = \mathbf{v} \circ F_K = \sum_{i=1}^5 N_i \begin{pmatrix} u_i \\ v_i \end{pmatrix} =: \begin{pmatrix} U_0 + U_1\xi + U_2\eta + U_{12}\xi\eta + U_{122}\xi\eta^2 \\ V_0 + V_1\xi + V_2\eta + V_{12}\xi\eta + V_{122}\xi\eta^2 \end{pmatrix}. \quad (3.23)$$

This yields

$$J_K \begin{pmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial v}{\partial y} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \end{pmatrix} = \begin{pmatrix} (U_1b_2 - U_2b_1) + (U_1b_{12} - U_{12}b_1)\xi + (U_{12}b_2 - U_2b_{12})\eta \\ \quad + U_{122}(b_2\eta^2 - b_{12}\xi\eta^2 - 2b_1\xi\eta) \\ (V_2a_1 - V_1a_2) + (V_{12}a_1 - V_1a_{12})\xi + (V_2a_{12} - V_{12}a_2)\eta \\ \quad + V_{122}(-a_2\eta^2 + a_{12}\xi\eta^2 + 2a_1\xi\eta) \\ (U_2a_1 - U_1a_2) + (U_{12}a_1 - U_1a_{12})\xi + (U_2a_{12} - U_{12}a_2)\eta + \\ \quad (V_1b_2 - V_2b_1) + (V_1b_{12} - V_{12}b_1)\xi + (V_{12}b_2 - V_2b_{12})\eta + \\ U_{122}(-a_2\eta^2 + a_{12}\xi\eta^2 + 2a_1\xi\eta) + V_{122}(b_2\eta^2 - b_{12}\xi\eta^2 - 2b_1\xi\eta) \end{pmatrix} \\ = \begin{pmatrix} b_2 + b_{12}\xi & 0 & -a_2 - a_{12}\xi \\ -b_1 - b_{12}\eta & 0 & a_1 + a_{12}\eta \\ -b_1\xi + b_2\eta & 0 & a_1\xi - a_2\eta \\ 0 & a_1 + a_{12}\eta & -b_1 - b_{12}\eta \\ 0 & a_1\xi - a_2\eta & -b_1\xi + b_2\eta \\ b_2\eta^2 - b_{12}\xi\eta^2 - 2b_1\xi\eta & 0 & -a_2\eta^2 + a_{12}\xi\eta^2 + 2a_1\xi\eta \\ 0 & -a_2\eta^2 + a_{12}\xi\eta^2 + 2a_1\xi\eta & b_2\eta^2 - b_{12}\xi\eta^2 - 2b_1\xi\eta \end{pmatrix}^T \beta^v \quad (3.24)$$

with

$$\beta^v = (\beta_1^v, \dots, \beta_7^v) := \left(U_1 + \frac{b_1}{a_1} V_1, U_2 + \frac{b_2}{a_1} V_1, U_{12} + \frac{b_{12}}{a_1} V_1, V_2 - \frac{a_2}{a_1} V_1, V_{12} - \frac{a_{12}}{a_1} V_1, U_{122}, V_{122} \right)^T.$$

Lemma 3.1. *For any $\mathbf{v} \in \mathbf{V}_h$ and $K \in \mathcal{T}_h$, it holds*

$$\|\epsilon(\mathbf{v})\|_{0,K}^2 \lesssim \frac{1}{\min_{(\xi,\eta) \in \hat{K}} J_K(\xi,\eta)} h_K^2 \sum_{1 \leq i \leq 7} (\beta_i^v)^2. \quad (3.25)$$

Proof. From (3.24) and (3.6), we have

$$\begin{aligned} \|\epsilon(\mathbf{v})\|_{0,K}^2 &= \int_K \epsilon(\mathbf{v}) : \epsilon(\mathbf{v}) d\mathbf{x} \\ &= \int_{\hat{K}} \left[((b_2 + b_{12}\xi)\beta_1^v - (b_1 + b_{12}\eta)\beta_2^v - (b_1\xi - b_2\eta)\beta_3^v + (b_2\eta^2 - b_{12}\xi\eta^2 - 2b_1\xi\eta)\beta_6^v)^2 \right. \\ &\quad + ((a_1 + a_{12}\eta)\beta_4^v + (a_1\xi - a_2\eta)\beta_5^v + (-a_2\eta^2 + a_{12}\xi\eta^2 + 2a_1\xi\eta)\beta_7^v)^2 \\ &\quad + \frac{1}{2}(-(a_2 + a_{12}\xi)\beta_1^v + (a_1 + a_{12}\eta)\beta_2^v + (a_1\xi - a_2\eta)\beta_3^v - (b_1 + b_{12}\eta)\beta_4^v - (b_1\xi - b_2\eta)\beta_5^v \\ &\quad \left. + (-a_2\eta^2 + a_{12}\xi\eta^2 + 2a_1\xi\eta)\beta_6^v + (b_2\eta^2 - b_{12}\xi\eta^2 - 2b_1\xi\eta)\beta_7^v)^2 \right] J_K^{-1}(\xi,\eta) d\xi d\eta \\ &\lesssim \frac{1}{\min_{(\xi,\eta) \in \hat{K}} J_K(\xi,\eta)} h_K^2 \sum_{1 \leq i \leq 7} (\beta_i^v)^2. \end{aligned} \quad \square$$

Lemma 3.2. *For any $\tau \in \Sigma_h$ and $K \in \mathcal{T}_h$, it holds that*

$$\|\tau\|_{0,K}^2 \gtrsim \min_{(\xi,\eta) \in \hat{K}} J_K(\xi,\eta) \sum_{1 \leq i \leq 7} (\beta_i^\tau)^2. \quad (3.26)$$

Proof. From (3.16) and (3.6), we have

$$\begin{aligned} \|\tau\|_{0,K}^2 &= \int_K \tau : \tau d\mathbf{x} \\ &= \int_{\hat{K}} \left[(\beta_1^\tau + \eta\beta_4^\tau + \xi\beta_6^\tau)^2 + (\beta_2^\tau + \xi\beta_5^\tau + \eta\beta_7^\tau)^2 + 2(\beta_3^\tau + \frac{b_1^2\xi + b_1b_2\eta}{J_0}\beta_4^\tau \right. \\ &\quad \left. + \frac{a_1a_2\xi + a_2^2\eta}{J_0}\beta_5^\tau - \frac{b_1b_2\xi + b_2^2\eta}{J_0}\beta_6^\tau - \frac{a_1^2\xi + a_1a_2\eta}{J_0}\beta_7^\tau)^2 \right] J_K(\xi,\eta) d\xi d\eta \\ &\gtrsim \min_{(\xi,\eta) \in \hat{K}} J_K(\xi,\eta) \sum_{1 \leq i \leq 7} (\beta_i^\tau)^2. \end{aligned} \quad \square$$

We introduce a mesh condition given by Shi [30]:

Condition (A). The distance $d_K (= 2\sqrt{a_{12}^2 + b_{12}^2})$ between the midpoints of the diagonals of $K \in \mathcal{T}_h$ (see Figure 3) is of order $o(h_K)$ uniformly for all elements K as $h \rightarrow 0$.

Under this condition we have

$$\max\{|a_{12}|, |b_{12}|\} = o(h_K). \quad (3.27)$$

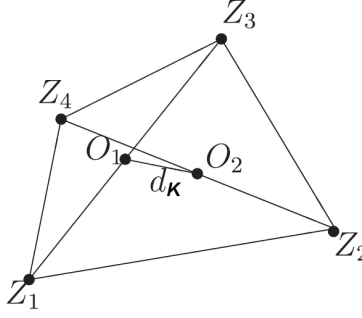


Figure 3: The distance d_K between midpoints of two diagonals

Lemma 3.3. *Under **Condition (A)**, for any $\mathbf{v} \in \mathbf{V}_h$ there exists a $\tau_v \in \Sigma_h$ such that for any $K \in \mathcal{T}_h$,*

$$\int_K \tau_v : \epsilon(\mathbf{v}) d\mathbf{x} = \|\tau_v\|_{0,K}^2 \gtrsim \|\epsilon(\mathbf{v})\|_{0,K}^2. \quad (3.28)$$

Proof. We follow the same line as in the proof of [35]. For $\tau \in \Sigma_h$ and $\mathbf{v} \in \mathbf{V}_h$, from (3.16) and (3.24), it holds that

$$\int_K \tau : \epsilon(\mathbf{v}) d\mathbf{x} = (\beta^\tau)^T A \beta^v,$$

where $A = (A_1 \ A_2)$ and

$$A_1 = \begin{pmatrix} 4b_2 & -4b_1 & 0 & 0 \\ 0 & 0 & 0 & 4a_1 \\ -4a_2 & 4a_1 & 0 & -4b_1 \\ -\frac{4a_{12}b_1^2}{3J_0} & \frac{4a_{12}b_1b_2-4b_{12}J_0}{3J_0} & \frac{4b_2J_0+4a_1b_1^2-4a_2b_2b_1}{3J_0} & -\frac{4b_1b_2b_{12}}{3J_0} \\ -\frac{4a_1a_2a_{12}}{3J_0} & \frac{4a_2^2a_{12}}{3J_0} & \frac{4a_2(a_1^2-a_2^2)}{3J_0} & -\frac{4a_2^2b_{12}}{3J_0} \\ \frac{4b_{12}J_0+4a_{12}b_1b_2}{3J_0} & -\frac{4a_{12}b_2^2}{3J_0} & \frac{4a_2b_2^2-4a_1b_1b_2-4b_1J_0}{3J_0} & \frac{4b_2^2b_{12}}{3J_0} \\ \frac{4a_1^2a_{12}}{3J_0} & -\frac{4a_1a_2a_{12}}{3J_0} & -\frac{4a_1(a_1^2-a_2^2)}{3J_0} & \frac{4a_{12}J_0+4a_1a_2b_{12}}{3J_0} \end{pmatrix},$$

$$A_2 = \begin{pmatrix} 0 & \frac{4b_2}{3} & 0 \\ 0 & 0 & -\frac{4a_2}{3} \\ 0 & -\frac{4a_2}{3} & \frac{4b_2}{3} \\ -\frac{4b_1(b_1^2-b_2^2)}{3J_0} & \frac{4a_{12}b_1^2}{9J_0} & -\frac{4b_1^2b_{12}}{9J_0} \\ \frac{4a_1J_0+4b_2a_2^2-4a_1b_1a_2}{3J_0} & \frac{4a_1a_2a_{12}}{9J_0} & \frac{4a_{12}J_0-4a_1a_2b_{12}}{9J_0} \\ \frac{4b_2(b_1^2-b_2^2)}{3J_0} & -\frac{4b_{12}J_0-4a_{12}b_1b_2}{9J_0} & \frac{4b_1b_2b_{12}}{9J_0} \\ \frac{4b_1a_1^2-4a_2b_2a_1-4a_2J_0}{3J_0} & -\frac{4a_1^2a_{12}}{9J_0} & \frac{4a_1^2b_{12}}{9J_0} \end{pmatrix}.$$

By the mean value theorem, there exists a point $(\xi_0, \eta_0) \in [-1, 1]^2$ such that

$$\|\tau\|_{0,K}^2 = J_k(\xi_0, \eta_0)(\beta^\tau)^T D \beta^\tau, \quad (3.29)$$

where

$$D = \begin{pmatrix} 4 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 8 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{8b_1^4+8b_1^2b_2^2+4J_0^2}{3J_0^2} & \frac{8a_2b_1(a_1b_1+a_2b_2)}{3J_0^2} & -\frac{8b_1b_2(b_1^2+b_2^2)}{3J_0^2} & -\frac{8a_1b_1(a_1b_1+a_2b_2)}{3J_0^2} \\ 0 & 0 & 0 & \frac{8a_2b_1(a_1b_1+a_2b_2)}{3J_0^2} & \frac{8a_1^2a_2^2+8a_2^4+4J_0^2}{3J_0^2} & -\frac{8a_2b_2(a_1b_1+a_2b_2)}{3J_0^2} & -\frac{8a_1a_2(a_1^2+a_2^2)}{3J_0^2} \\ 0 & 0 & 0 & -\frac{8b_1b_2(b_1^2+b_2^2)}{3J_0^2} & -\frac{8a_2b_2(a_1b_1+a_2b_2)}{3J_0^2} & \frac{8b_1^2b_2^2+8b_2^4+4J_0^2}{3J_0^2} & \frac{8a_1b_2(a_1b_1+a_2b_2)}{3J_0^2} \\ 0 & 0 & 0 & -\frac{8a_1b_1(a_1b_1+a_2b_2)}{3J_0^2} & -\frac{8a_1a_2(a_1^2+a_2^2)}{3J_0^2} & \frac{8a_1b_2(a_1b_1+a_2b_2)}{3J_0^2} & \frac{8a_1^4+8a_1^2a_2^2+4J_0^2}{3J_0^2} \end{pmatrix}.$$

By taking

$$\tau = \begin{pmatrix} 1 & 0 & 0 & \eta & 0 & \xi & 0 \\ 0 & 1 & 0 & 0 & \xi & 0 & \eta \\ 0 & 0 & 1 & \frac{b_1^2\xi+b_1b_2\eta}{a_1b_2-a_2b_1} & \frac{a_1a_2\xi+a_2^2\eta}{a_1b_2-a_2b_1} & \frac{b_1b_2\xi+b_2^2\eta}{a_2b_1-a_1b_2} & \frac{a_1^2\xi+a_1a_2\eta}{a_2b_1-a_1b_2} \end{pmatrix} \beta^{\tau,v}$$

with

$$\beta^{\tau,v} = \frac{1}{J_K(\xi, \eta)} D^{-1} A \beta^v, \quad (3.30)$$

we immediately obtain

$$\int_K \tau_v : \epsilon(\mathbf{v}) d\mathbf{x} = \|\tau_v\|_{0,K}^2 \quad (3.31)$$

and

$$\beta^v = J_K(\xi_0, \eta_0) A^{-1} D \beta^{\tau,v}.$$

From **Condition (A)** and (3.6), we see that each entry of A is $O(\frac{1}{h})$ and each entry of D is $O(1)$, which implies

$$\sum_{1 \leq i \leq 7} (\beta_i^v)^2 \lesssim h_K^2 \sum_{1 \leq i \leq 7} (\beta_i^{\tau,v})^2.$$

Combining this inequality with Lemmas 3.1–3.2 and (3.6), we obtain

$$\|\tau_v\|_{0,K}^2 \gtrsim \|\epsilon(\mathbf{v})\|_{0,K}^2. \quad \square$$

Remark 3.6. It has been shown in [35] that Lemma 3.3 holds when K is a 4-node quadrilateral, which is corresponding to the hybrid stress elements PS [24] or ECQ4 [32].

Proposition 3.2. Let the partition \mathcal{T}_h satisfy the shape-regularity condition (3.1) and **Condition (A)**, then the uniform discrete inf-sup condition $(A2_h)$ holds for the 5-node hybrid stress transition element.

Proof. We can get the desired conclusion by following the same line as in the proof of Theorem 4.2 in [35]. In fact, from Lemma 3.3, for any $\mathbf{v} \in \mathbf{V}_h$ there exists some $\tau_v \in \Sigma_h$ such that (3.28) holds. This means

$$\|\tau_v\|_0 \|\mathbf{v}\|_h \lesssim \left(\sum_K \int_K \tau_v : \tau_v d\mathbf{x} \right)^{1/2} \left(\sum_K \int_K \epsilon(\mathbf{v}) : \epsilon(\mathbf{v}) d\mathbf{x} \right)^{1/2} \lesssim \sum_K \int_K \tau_v : \epsilon(\mathbf{v}) d\mathbf{x}.$$

Then the stability $(A2_h)$ follows immediately. \square

Combining Propositions 3.1–3.2 and the standard theory of mixed finite element methods (cf. [15]), we have the following uniform estimate for the 5-node hybrid stress transition element:

Theorem 3.1. *Let $(\sigma, \mathbf{u}) \in \Sigma \times \mathbf{V}$ be the solution of the variational problem (2.1)–(2.2). Under the conditions of Propositions 3.1–3.2, the discretization problem (3.21)–(3.22) admits a unique solution $(\sigma_h, \mathbf{u}_h) \in \Sigma_h \times \mathbf{V}_h$ such that*

$$\|\sigma - \sigma_h\|_0 + \|\mathbf{u} - \mathbf{u}_h\|_h \lesssim \inf_{\tau \in \Sigma_h} \|\sigma - \tau\|_0 + \inf_{\mathbf{v} \in \mathbf{V}_h} \|\mathbf{u} - \mathbf{v}\|_h + \sup_{\mathbf{w} \in \mathbf{V}_h \setminus \{0\}} \frac{|b(\sigma, \mathbf{w}) - b_h(\sigma, \mathbf{w})|}{\|\mathbf{w}\|_h} \quad (3.32)$$

For the consistency error term in the estimate (3.32), we have

$$|b(\sigma, \mathbf{w}) - b_h(\sigma, \mathbf{w})| = |(-\operatorname{div} \sigma, \mathbf{w}) - b_h(\sigma, \mathbf{w})| = \left| \sum_{e \in \mathcal{E}_h^*} \int_e \sigma \mathbf{n}_e \cdot [\mathbf{w}] dx \right|, \quad (3.33)$$

where \mathbf{n}_e is the unit outer normal vector along e . The work left to us is to estimate (3.33).

Let \mathcal{T}_h^* be the set of all marco-elements, like \tilde{K} in Figure 4, of \mathcal{T}_h . Following the same

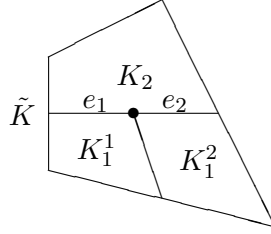


Figure 4: micro-element \tilde{K}

procedure as in [17], we have the following estimate for the consistency error term.

Lemma 3.4. *It holds that*

$$\sup_{\mathbf{w} \in \mathbf{V}_h \setminus \{0\}} \frac{|b(\sigma, \mathbf{w}) - b_h(\sigma, \mathbf{w})|}{\|\mathbf{w}\|_h} \leq h |\sigma|_{1, \mathcal{T}_h^*}, \quad (3.34)$$

where $|\sigma|_{1, \mathcal{T}_h^*} := \left(\sum_{\tilde{K} \in \mathcal{T}_h^*} |\sigma|_{1, \tilde{K}}^2 \right)^{1/2}$.

Proof. As shown in Figure 4, we denote

$$K_1 := K_1^1 \cup K_1^2, \quad \tilde{K} := K_1 \cup K_2, \quad (3.35)$$

$$e := e_1 \cup e_2 \text{ with } e_i := \overline{K_1^i} \cap \overline{K_2} \text{ for } i = 1, 2,$$

$$\mathbf{w}_i^e = (\mathbf{w}|_{K_i})|_e \text{ for } i = 1, 2, \quad \bar{\mathbf{w}}^e = \frac{1}{|\tilde{e}|} \int_e \mathbf{w}_1^e dx.$$

By (3.13) it also holds $\bar{\mathbf{w}}^e = \frac{1}{|\tilde{e}|} \int_e \mathbf{w}_2^e dx$. Standard scaling arguments, together with trace inequality, yields

$$|\mathbf{w} - \bar{\mathbf{w}}^e|_e \lesssim h^{1/2} |\mathbf{w}|_{1, \tilde{K}}. \quad (3.36)$$

For any $\zeta \in H^1(\tilde{K})^2$, from trace inequality and Poincaré inequality it follows

$$|\zeta - \bar{\zeta}|_{0,e} \lesssim h^{1/2} |\zeta|_{1,\tilde{K}}, \quad (3.37)$$

where $\bar{\zeta} := \frac{1}{|\tilde{K}|} \int_{\tilde{K}} \zeta dx$. The estimates (3.36)–(3.37), together with (3.13), imply

$$\left| \int_e \zeta [\mathbf{w}] ds \right| = \left| \int_e (\zeta - \bar{\zeta}) [\mathbf{w} - \bar{\mathbf{w}}^e] ds \right| \lesssim h |\zeta|_{1,\tilde{K}} |\mathbf{w}|_{1,\tilde{K}}.$$

Taking $\zeta = \sigma \mathbf{n}_e$ in the above inequality and summing over all $e \in \mathcal{E}_h^*$, we obtain

$$\left| \sum_{e \in \mathcal{E}_h^*} \int_e \sigma \mathbf{n}_e [\mathbf{w}] ds \right| \lesssim \sum_{\tilde{K} \in \mathcal{T}_h^*} h |\sigma|_{1,\tilde{K}} |\mathbf{w}|_{1,\tilde{K}} \leq h |\sigma|_{1,\mathcal{T}_h^*} \|\mathbf{w}\|_h,$$

□

which yields the desired result (3.34).

From Theorem 3.1, Lemma 3.4, and the standard interpolation theory, we have the following uniform a priori error estimation.

Theorem 3.2. *Let $(\sigma, \mathbf{u}) \in \Sigma \cap H^1(\Omega; \mathbb{R}_{sym}^{2 \times 2}) \times \mathbf{V} \cap H^2(\Omega)^2$ and $(\sigma_h, \mathbf{u}_h) \in \Sigma_h \times \mathbf{V}_h$ be respectively the solutions of the weak problem (2.1)–(2.2) and of the discretized problem (3.21)–(3.22). Under the same assumptions of Theorem 3.1 it holds*

$$\|\sigma - \sigma_h\|_0 + \|\mathbf{u} - \mathbf{u}_h\|_h \lesssim h(|\sigma|_1 + |\mathbf{u}|_2).$$

4 An adaptive algorithm for quadrilateral meshes

Inspired by the coarsening algorithm in [9], we introduce new refinement/coarsening algorithms for quadrilateral meshes. Unlike the classical recursive refinement/coarsening procedures, the proposed algorithms are non-recursive and require neither storing nor maintaining refinement tree information such as the parents, brothers, generation, etc. The main idea is using a special ordering of the elements in the data structure. This also makes the implementation easier. We note that the algorithms for quadrilateral meshes are considerably more complicated than their triangular counterparts owing to the existence of hanging nodes.

4.1 Data structures

Our basic data structure for quadrilateral meshes contains five arrays, `node(1:N,1:2)`, `node_flag(1:N,1)`, `edge(1:NE,1:2)`, `edge_flag(1:NE,1)`, and `elem(1:NT,1:12)`, where `N` is the number of vertices, `NE` is the number of edges, and `NT` is the number of elements.

In the node array `node`, the first and second columns contain x – and y –coordinates of the nodes in the mesh; see Table 1. In the `node_flag`, it contains the flags for the nodes: ‘0’ for regular nodes, ‘-1’ for “newest” nodes, ‘-2’ for boundary nodes (one could define more flags such as Dirichlet Boundary, Neumann Boundary etc). A “newest” node refers to the internal

point generated by a refinement of a quadrilateral element. In the edge array **edge**, the two columns contain indices to the vertices of the edge; see Table 2. In **edge_flag**, the only column contains the flags for the edges: ‘-2’ for boundary edge, ‘0’ for regular edge, ‘ $2e_f$ ’ if the index of this edge is bigger than its brother, ‘ $2e_f - 1$ ’-if index of the edge is smaller than its brother, where ‘ e_c ’ is the index of “father” edge . In the element array **elem**, the first four columns contain indices to the vertices of elements, the 5-8th columns contain indices to the mid-nodes of edges of elements (‘0’-if there is no mid-node), and the 9-12th columns contains the edges of elements.

Remark 4.1. As an example, **node**, **node_flag**, **edge**, **edge_flag**, and **elem** matrices to represent a triangulation of the L-shaped domain $\Omega = (-1, -1) \times (1, 1) \setminus ([-1, 0] \times [-1, 0])$ are given in Figure 5 and Table 1–3.

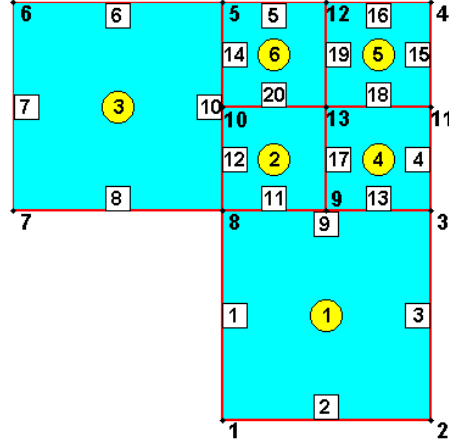


Figure 5: Quadrilateral mesh of L-shaped domain

x	0	1	1	1	0	-1	-1	0	0.5	0	1	0.5	0.5
y	-1	-1	0	1	1	1	0	0	0	0.5	0.5	1	0.5
flag	-2	-2	-2	-2	-2	-2	-2	-2	0	0	-2	-2	-1
	1	2	3	4	5	6	7	8	9	10	11	12	13

Table 1: **node** & **node_flag** : coordinates and flag for each node

node ₁	1	1	2	3	5	6	7	7	8	8	8	8	9	10	11	12	9	13	13	10
node ₂	8	2	3	11	12	5	6	8	3	5	9	10	3	5	4	4	13	11	12	13
flag	-2	-2	-2	-2	-2	-2	-2	-2	0	0	17	19	18	20	-2	-2	0	0	0	0
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

Table 2: **edge** & **edge_flag** : edge end-point indices and edge flags

For convenience of implementation, we also introduce two auxiliary arrays: **edge2elem** and **node2elem**. **edge2elem** (Table 4) is a sparse matrix in IJ-format, whose rows and columns denote the indices of elements and edges, respectively. The (i, j) -entry of the matrix denotes the local index of the j -th **edge** in the i -th **elem**. **node2elem** (Table 5) is a sparse pattern of a

elem ₁	1	2	3	8	0	0	9	0	2	3	9	1
elem ₂	8	9	13	10	0	0	0	0	11	17	20	12
elem ₃	7	8	5	6	0	10	0	0	8	10	6	7
elem ₄	9	3	11	13	0	0	0	0	13	4	18	17
elem ₅	13	11	4	12	0	0	0	0	18	15	16	19
elem ₆	10	13	12	5	0	0	0	0	20	19	5	14
	n ₁	n ₂	n ₃	n ₄	n ₅	n ₆	n ₇	n ₈	e ₁	e ₂	e ₃	e ₄

Table 3: **elem** : node and edge indices of each element

sparse matrix in IJ-format, whose rows and columns denote the indices of elements and nodes respectively.

If an element contains a certain edge/node, we say that this element is an adjacent element of the edge/node. We now define “good-for-coarsening” or “good” node as a newest node whose adjacent elements have no hanging node. In other word, the **node_flag** for a “good” node is ‘-1’, and the **n_5,...,n_8** (in **node2elem**) are all ‘0’ for its adjacent elements.

1	1	1	4	6	3	3	3	1	3	2	2	4	6	5	5	2	4	4	5	5	6	2	6
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	17	18	18	19	19	20	20
4	1	2	2	3	3	4	1	3	2	1	4	1	4	2	3	2	4	3	1	4	2	3	1

Table 4: **edge2elem** : the first two rows denote the indices of elements and edges, respectively; the third row denotes the corresponding local indices of the edges in the element

1	1	1	4	5	3	6	3	3	1	2	3	2	4	2	6	4	5	5	6	2	4	5	6
1	2	3	3	4	5	5	6	7	8	8	8	9	9	10	10	11	11	12	12	13	13	13	13

Table 5: **node2elem** : indices of elements and nodes

4.2 Refinement and coarsening algorithms

Due to the fact that we are going to implement the algorithms in Matlab, we avoid to perform refinement/coarsening element by element. Instead, we mark edges of all marked element, categorize these edges, and perform vectorized operations for each case. Our algorithms work as follows:

$$\mathbf{mesh} + \mathbf{indices\ of\ marked\ elem} \xrightarrow{\text{refine/coarsen}} \mathbf{new\ mesh} .$$

Before refining a quadrilateral mesh, we need a post-marking step in order to make sure that there will be no more than one hanging node on each edge after the refinement. We use **edge_m** to denote the indices of marked edges, and **elem_m** to denote the indices of marked elements. In this post-marking procedure, we first get **edge_m** from **elem_m**. Then we find the edges with “hanging” node from **edge_m**, and we use **edge_hg** to denote the indices of these edges. Finally, based on **edge2elem**, we find all the elements who contain **edge_hg**. By adding these elements to **elem_m** we get a new **elem_m**. If the new and old **elem_m** are the same, then the post-marking procedure terminates, otherwise we do this procedure iteratively.

Now we categorize marked edges (for refinement or coarsening) into several different types. We use **elem_adj** to denote the neighboring element(s) of marked edges, **elem2remove** to denote the elements which will be removed by coarsening.

(1) For refinement

Type 1: no hanging node and `edge_flag` = 0 belongs to one `edge_m`.

Type 2: no hanging node and `edge_flag` = 0 belongs to two `edge_m` or on the boundary.

Type 3: no hanging node and `edge_flag` > 0 is an odd number.

Type 4: no hanging node and `edge_flag` > 0 is an even number.

Type 5: with one hanging node.

(2) For coarsening

Type 1: has two `elem_adj`, and only one of them belongs to `elem2remove`.

Type 2: has one `elem_adj`, but not on the boundary.

Type 3: has two `elem_adj`, and both of them belong to `elem2remove` or on the boundary.

The algorithm for refinement/coarsening can be found in **Algorithms 1–2**. To make the algorithms more accessible by readers, we use the mesh in Figure 5 as an example to explain the edge types. Let e_i be the i -th `edge` and E_i be the i -th `elem`. For the refinement algorithm, we can see that

- If E_6 is marked to be refined, but not for E_5 , then e_{19} belongs to Type 1.
- If E_6 and E_5 are both marked to be refined, then e_{19} belongs to Type 2.
- If E_2 is marked to be refined, then e_{12} belongs to Type 3.
- If E_6 is marked to be refined, then e_{14} belongs to Type 4.
- If E_3 is marked to be refined, then e_7 belongs to Type 2 and e_{10} belongs to Type 5.

On the other hand, for the coarsen procedure, we have

- If E_6 is marked for coarsening, but not for E_5 , then e_{19} belongs to Type 1.
- If E_6 is marked for coarsening, but not for E_3 , then e_{14} belongs to Type 2.
- If E_6 and E_5 are both marked for coarsening, then e_{19} belongs to Type 3.
- If E_3 is marked for coarsening, then e_7 belongs to Type 3.

Now we present the algorithms for refinement and coarsening.

4.3 Adaptive algorithm

We are now ready to present the adaptive algorithm for discrete problem (3.21)–(3.22) with the transition hybrid stress element. The adaptive algorithm is given in **Algorithm 3**.

Algorithm 1 REFINE(*mesh*, *elem.m*)

1. Categorize the edges need to be refined and save as *edge.m*.
 2. Update mesh info based on the type of *edge.m*:
 - Type 1: add one node, two edges, and update adjacent element.
 - Type 2: add one node, one edge.
 - Type 3: add one node, two edges, and update *edge_flag* information.
 - Type 4: add one node, two edges, and update *edge_flag* information.
 - Type 5: update the elements in the patch of the edge.
-

Algorithm 2 COARSEN(*mesh*, *elem.m*)

1. Find “good” nodes.
 2. Mark all elements who contain the “good” nodes as *elem2remove*.
 3. Categorize all the edges of *elem2remove*.
 4. Update mesh info based on the type
 - Type 1 : add one new edge, update *edge* info for edges belong to Type 1 & 2.
 - Type 2 : remove one node, update *elem* info for corresponding *elem_adj*.
 - Type 3 : remove one node, update *edge* info for edges belong to Type 3.
-

Algorithm 3 AFEM

FOR $l = 0, 1, 2, \dots$ UNTIL termination on level L , DO:

1. Solve the discrete problem (3.21)–(3.22) on \mathcal{T}_l ;
 2. Compute $\eta_N := (\sum_{K \in \mathcal{T}_l} \eta_K^2)^{1/2}$ with $\eta_K := |H_K|^{1/2}|K|, \forall K \in \mathcal{T}_l$ as error indicators [8],
where $H_K = \frac{1}{|K|} \int_K \text{diag}(\text{abs}(\text{svd}(\nabla \sigma_{h,K}))) dx + 10^{-8} I$;
 3. Mark a set of elements M_l in \mathcal{T}_l with minimal cardinality such that $\sum_{K \in M_l} \eta_K^2 > \frac{1}{2} \eta_N^2$;
 4. Refine \mathcal{T}_l to obtain \mathcal{T}_{l+1} .
-

5 Numerical examples

In the following numerical examples, we use MATLAB (R2011a) to implement the algorithms, and the experimental platform is a desktop with Intel Xeon E5640@2.67GHz CPU and CentOS 6.5.

5.1 Poisson’s equation on an L -shaped domain

The domain is as in Figure 5, where $\Omega = [-1, 1]^2 \setminus [-1, 0]^2$, $u = r^{\frac{2}{3}} \sin((2\theta + \pi)/3)$ (polar coordinate), $\Gamma_D = \partial\Omega$, $f = 0$. We call the standard h-refinement adaptive method [14] to solve the Poisson problem with the refinement algorithm proposed in the previous section for quadrilateral meshes. And we compare its performance with the bisection algorithm for triangular

meshes in iFEM [7]. Figures 6–7 and Table 6 compare the two algorithms. For Table 6, we start from initial meshes of the same mesh size, and run the adaptive algorithm until the error $|u - u_h|_1 < 10^{-3}$. Here and in what following, we use DOF to denote the degree of freedom.

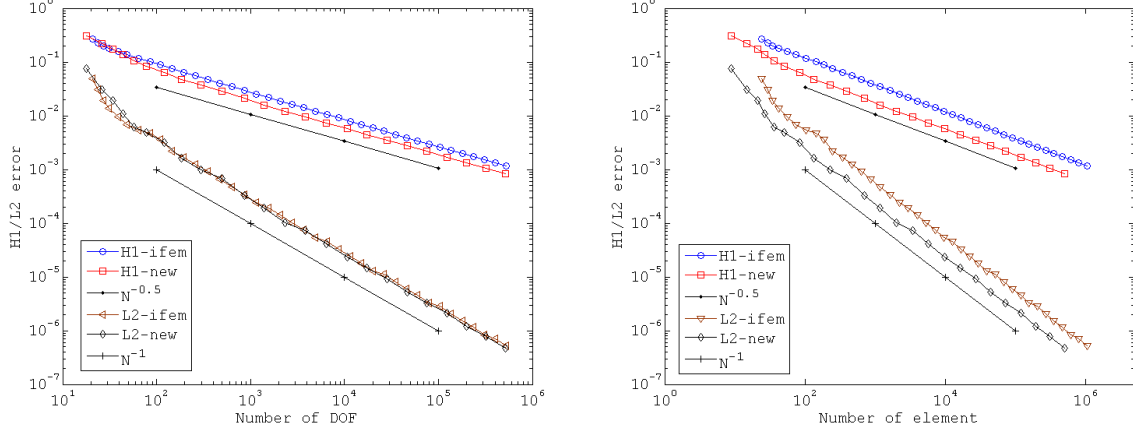


Figure 6: Convergence rate of h-refinement for the Poisson's equation

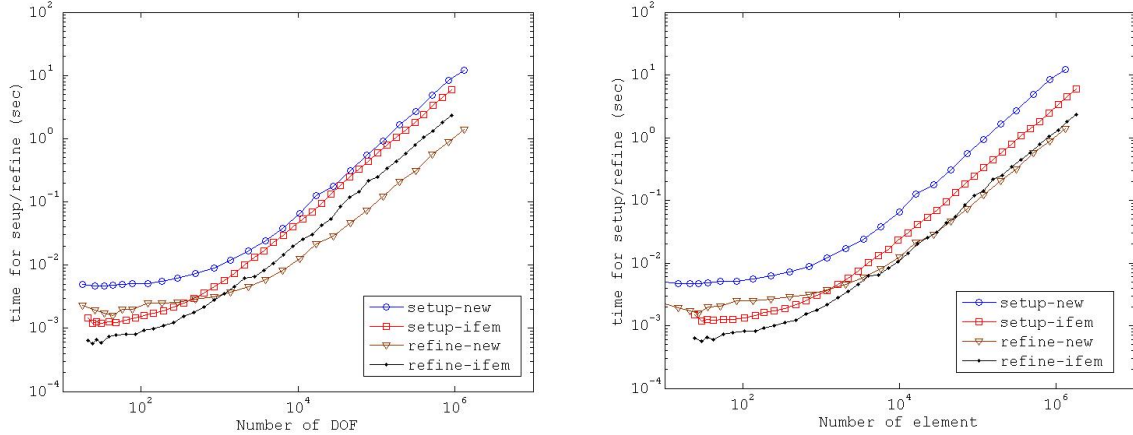


Figure 7: Computation time (seconds) of h-refinement for the Poisson's equation

	Final mesh DOF	$ u - u_h _1$	Total time (sec)
iFEM	697322	9.97×10^{-4}	79.0509
NEW	517433	8.31×10^{-4}	35.2750

Table 6: Performance of h-refinement algorithms for the Poisson's equation

5.2 Moving circle

This example is used to test the performance of refinement and coarsening. What we want to do is to track the interface of $x^2 + y^2 = (0.5 - t)^2, t \in [0, 1]$. One of the tracking state is given

in Figure 8 and the performance of the new refinement and coarsening algorithms are given in Figure 9.

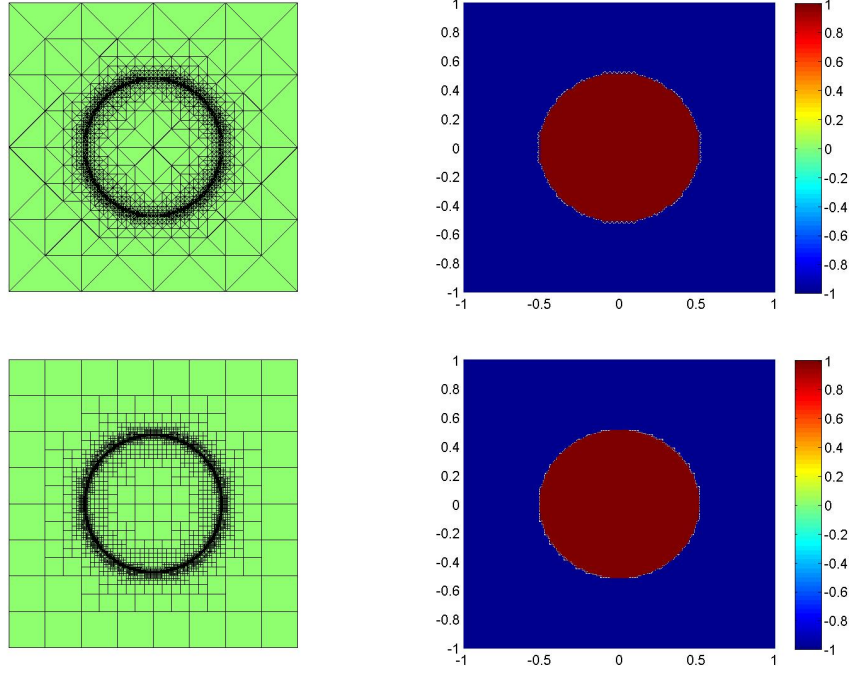


Figure 8: Tracking state for moving circle, Left: mesh; Right: interface; Upper: triangular; Lower: quadrilateral

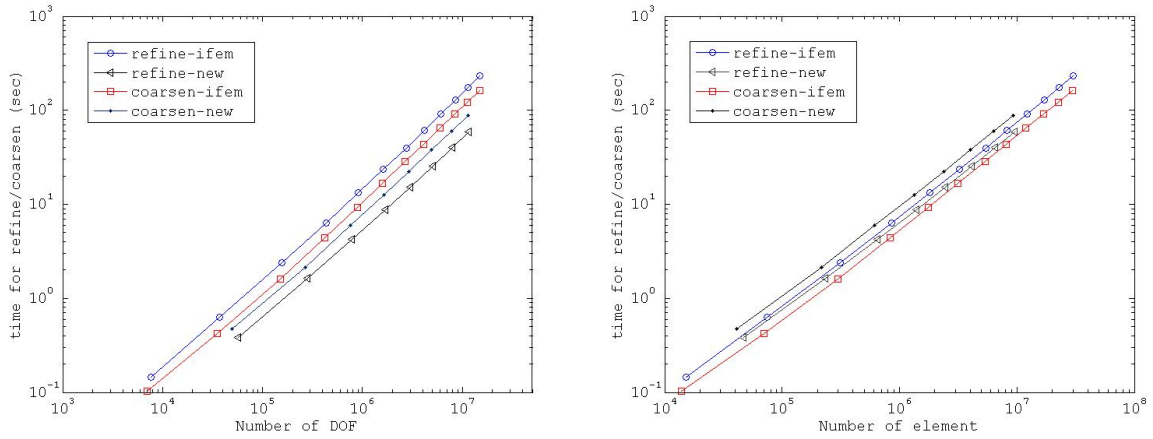


Figure 9: Computation time (seconds) for the moving circle test

The results of the above two examples in §5.1–5.2, suggest that

- Convergence rate of the new adaptive algorithm is optimal (same as iFEM), while the errors by the new algorithm are slightly smaller than the ones by iFEM; Furthermore, to

reach same level of accuracy for solving the Poisson's equation, the adaptive quadrilateral meshes costs less computational time than the corresponding adaptive bisection meshes.

- The new algorithms demonstrate experimentally linear computational complexity (as efficient as iFEM) in both refinement and coarsening.

5.3 Poisson's ratio locking-free tests

Two test problems are used to examine locking-free performance of the 5-node transition hybrid stress element.

The first one, a plane strain pure bending cantilever beam (Figures 10–11), is a benchmark test widely used in the literature, e.g. [24, 25, 23, 32, 33, 35, 38]. The origin of the coordinates x, y is at the midpoint of the left end. The body force $\mathbf{f} = (0, 0)^T$, the surface traction \mathbf{g} defined on $\Gamma_N = \{(x, y) \in [0, 10] \times [-1, 1] : x = 10 \text{ or } y = \pm 1\}$ is given by $\mathbf{g}|_{x=10} = (-2Ey, 0)^T$, $\mathbf{g}|_{y=\pm 1} = (0, 0)^T$, and the exact solution is [35]

$$\mathbf{u} = \begin{pmatrix} -2(1-\nu^2)xy \\ (1-\nu^2)x^2 + \nu(1+\nu)(y^2-1) \end{pmatrix}, \quad \sigma = \begin{pmatrix} -2Ey & 0 \\ 0 & 0 \end{pmatrix}. \quad (5.1)$$



Figure 10: regular meshes

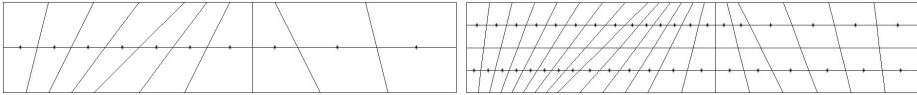


Figure 11: irregular meshes

The numerical results with $E = 1500$ and different values of Poisson's ratio ν are listed in Tables 7–8. The hybrid stress transition element gives uniformly good results as Poisson's ratio $\nu \rightarrow 0.5$ or Lamé constant $\lambda \rightarrow \infty$, with first order accuracy for the displacement approximation and more than first order accuracy for the stress approximation.

In the test example above, the stress approximation is very accurate. This is partially owing to the fact that the analytical stress solution is a linear-polynomial tensor. We now use a more difficult plane strain test with the same domain and initial meshes as in Figures 10–11. In this test, $\Gamma_N = \{(x, y) \in [0, 10] \times [-1, 1] : x = 10 \text{ or } y = \pm 1\}$, $\mathbf{f} = 12(\frac{x^2(1-\nu)+y^2\nu}{1-\nu^2}, 0)^T$, $\mathbf{g}|_{x=10} = (\frac{-4000(1-\nu)-120y^2\nu}{1-\nu^2}, 0)^T$, $\mathbf{g}|_{y=\pm 1} = (0, 0)^T$. The exact displacement and stress solutions are known as

$$\mathbf{u} = \frac{1}{E} \begin{pmatrix} -x^4(1-\nu) - 6x^2y^2\nu - \frac{y^4\nu^2}{1-\nu} \\ 4x^3y\nu + \frac{4xy^3\nu^2}{1-\nu} \end{pmatrix}, \quad \sigma = \begin{pmatrix} \frac{-4x^3(1-\nu)-12xy^2\nu}{1-\nu^2} & 0 \\ 0 & 0 \end{pmatrix}. \quad (5.2)$$

ν	regular		meshes		irregular		meshes	
	10×2	20×4	40×8	80×16	10×2	20×4	40×8	80×16
0.49	0.0478	0.0239	0.0120	0.0060	0.1033	0.0530	0.0268	0.0134
0.499	0.0496	0.0248	0.0124	0.0062	0.1047	0.0537	0.0271	0.0136
0.4999	0.0497	0.0249	0.0124	0.0062	0.1048	0.0538	0.0272	0.0136
0.49999	0.0497	0.0249	0.0124	0.0062	0.1048	0.0538	0.0272	0.0136
0.4999999999	0.0498	0.0249	0.0124	0.0062	0.1048	0.0538	0.0272	0.0136

Table 7: $\frac{\|\mathbf{u}-\mathbf{u}_h\|_h}{|\mathbf{u}|_1}$ for locking-free test 1

ν	regular		meshes		irregular		meshes	
	10×2	20×4	40×8	80×16	10×2	20×4	40×8	80×16
0.49	1.5e-3	7.1e-4	3.3e-4	9.4e-5	0.1018	0.0404	0.0149	0.0055
0.499	2.4e-4	7.2e-5	2.7e-5	9.9e-6	0.1022	0.0419	0.0159	0.0060
0.4999	1.6e-5	7.2e-6	2.7e-6	9.9e-7	0.1023	0.0421	0.0160	0.0060
0.49999	1.6e-6	7.2e-7	2.7e-7	9.9e-8	0.1023	0.0421	0.0160	0.0060
0.4999999999	0	0	0	0	0.1023	0.0421	0.0160	0.0060

Table 8: $\frac{\|\sigma-\sigma_h\|_0}{\|\sigma\|_0}$ for locking-free test 1

Numerical results in Tables 9–10 show that the hybrid stress transition element gives uniformly first order accuracy for both the displacement and stress approximations as the Poisson’s ratio $\nu \rightarrow 0.5$. This is exactly what we can expect from the theory.

ν	regular		meshes		irregular		meshes	
	10×2	20×4	40×8	80×16	10×2	20×4	40×8	80×16
0.49	0.1443	0.0720	0.0360	0.0180	0.1277	0.0632	0.0315	0.0157
0.499	0.1433	0.0716	0.0357	0.0179	0.1268	0.0628	0.0313	0.0156
0.4999	0.1432	0.0715	0.0357	0.0179	0.1267	0.0628	0.0313	0.0156
0.49999	0.1432	0.0715	0.0357	0.0179	0.1267	0.0628	0.0313	0.0156
0.4999999999	0.1432	0.0715	0.0357	0.0179	0.1267	0.0628	0.0313	0.0156

Table 9: $\frac{\|\mathbf{u}-\mathbf{u}_h\|_h}{|\mathbf{u}|_1}$ for locking-free test 2

5.4 Adaptive algorithm test with transition hybrid stress elements

We consider a square panel with edge length 2 and a one unit long edge crack [31]. Owing to symmetry, only the upper half of the panel is analyzed; see Figure 12. Along the positive x -axis, the condition of symmetry is applied, and on other edges, traction boundary conditions

ν	regular meshes				irregular meshes			
	10×2	20×4	40×8	80×16	10×2	20×4	40×8	80×16
0.49	0.0518	0.0256	0.0127	0.0064	0.0527	0.0258	0.0128	0.0064
0.499	0.0518	0.0256	0.0127	0.0064	0.0528	0.0258	0.0128	0.0064
0.4999	0.0518	0.0256	0.0127	0.0064	0.0528	0.0258	0.0128	0.0064
0.49999	0.0518	0.0256	0.0127	0.0064	0.0528	0.0258	0.0128	0.0064
0.4999999999	0.0518	0.0256	0.0127	0.0064	0.0528	0.0258	0.0128	0.0064

Table 10: $\frac{\|\sigma - \sigma_h\|_0}{\|\sigma\|_0}$ for locking-free test 2

are prescribed according to the following mode I crack solution in polar coordinate [1]:

$$\sigma = \frac{1}{\sqrt{r}} \left\{ \cos \frac{\theta}{2} \left(1 - \sin \frac{\theta}{2} \sin \frac{3\theta}{2} \right), \cos \frac{\theta}{2} \left(1 + \sin \frac{\theta}{2} \sin \frac{3\theta}{2} \right), \sin \frac{\theta}{2} \cos \frac{\theta}{2} \cos \frac{3\theta}{2} \right\}.$$

The $\frac{1}{\sqrt{r}}$ stress singularity occurs at the crack tip.

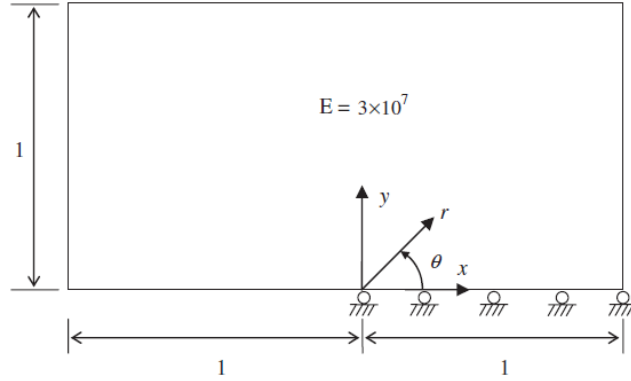


Figure 12: Schematic diagram for half of a cracked plane strain panel. Along $x = \pm 1$ and $y = 1$, exact tractions are prescribed.

A 8×4 uniform mesh is taken as the initial mesh. We show the relation between the number of DOF and the relative error $\frac{\|\sigma - \sigma_h\|}{\|\sigma\|}$ in Figure 13. We can see that the stress error uniformly reduces with a fixed factor on two successive meshes, and that the error on the adaptively refined meshes decreases more rapidly than the one on the uniformly refined meshes.

Acknowledgments. We would like to thank Prof. Pengtao Sun for his help on suggesting the error indicator used in the paper. Huang is supported by the China Scholarship Council. Xie is partially supported by the National Natural Science Foundation of China (11171239) and the Open Fund of Key Laboratory of Mountain Hazards and Earth Surface Processes, CAS. Zhang is partially supported by the NSFC Grant 91130011 and the National High Technology Research and Development Program of China Grant 2012AA01A309.

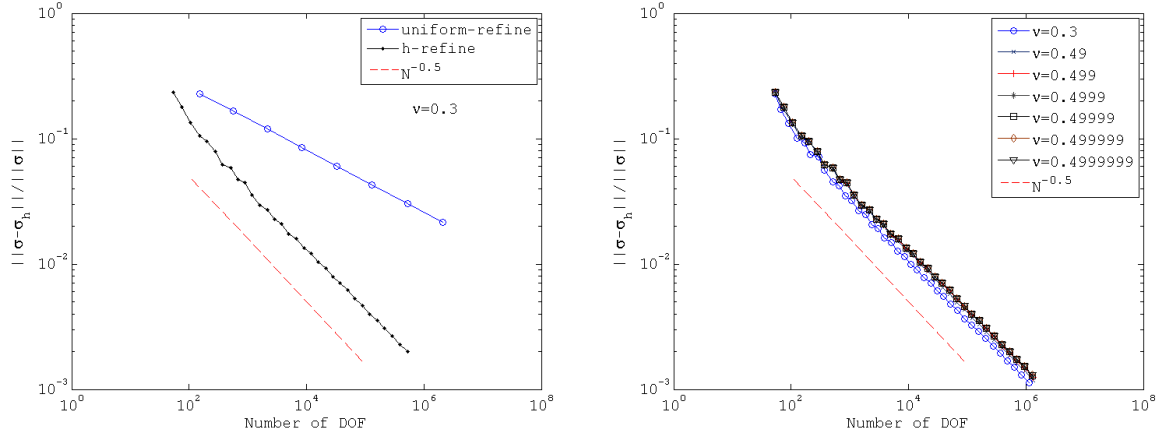


Figure 13: Convergence rates of adaptive hybrid stress transition elements for crack problem

References

- [1] Satya N Atluri. *Computational methods in the mechanics of fracture*, volume 2. North-Holland Amsterdam, 1986.
- [2] Wolfgang Bangerth, Ralf Hartmann, and Guido Kanschat. deal.II—a general-purpose object-oriented finite element library. *ACM Transactions on Mathematical Software*, 33(4):24, 2007.
- [3] Sören Bartels and Patrick Schreier. Local coarsening of simplicial finite element meshes generated by bisections. *BIT Numerical Mathematics*, 52(3):559–569, 2012.
- [4] Houman Borouchaki and Pascal J Frey. Adaptive triangular-quadrilateral mesh generation. *International Journal for Numerical Methods in Engineering*, 41:915–934, 1998.
- [5] Franco Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from lagrangian multipliers. *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique*, 8(R2):129–151, 1974.
- [6] Carsten Carstensen and Jun Hu. Hanging nodes in the unifying theory of a posteriori finite element error control. *Journal of Computational Mathematics*, 27(2-3):215–236, 2009.
- [7] Long Chen. iFEM: An innovative finite element method package in matlab. Technical report, University of California at Irvine, 2009.
- [8] Long Chen, Pengtao Sun, and Jinchao Xu. Optimal anisotropic meshes for minimizing interpolation errors in L^p -norm. *Mathematics of Computation*, 76(257):179–204, 2007.
- [9] Long Chen and Chen-Song Zhang. A coarsening algorithm on adaptive grids by newest vertex bisection and its applications. *Journal of Computational Mathematics*, 28(6):767–789, 2010.
- [10] Chang-Koon Choi and Eun-Jin Lee. Nonconforming variable-node axisymmetric solid element. *Journal of engineering mechanics*, 130(5):578–588, 2004.
- [11] Chang-Koon Choi and Nam-Ho Lee. Three dimensional transition solid elements for adaptive mesh gradation. *Structural Engineering and Mechanics*, 1(1):61–74, 1993.
- [12] Chang-Koon Choi and Yong-Myung Park. Nonconforming transition plate bending elements with variable mid-side nodes. *Computers & structures*, 32(2):295–304, 1989.
- [13] Chang-Koon Choi and Yong-Myung Park. Conforming and nonconforming transition plate bending elements for an adaptive h-refinement. *Thin-walled structures*, 28(1):1–20, 1997.

- [14] Willy Dörfler. A convergent adaptive algorithm for Poisson’s equation. *SIAM Journal on Numerical Analysis*, 33(3):1106–1124, 1996.
- [15] Michel Fortin and Franco Brezzi. Mixed and hybrid finite element methods. *Springer Series in Computational Mathematics*, 15, 1991.
- [16] Ajaya Kumar Gupta. A finite element for transition from a fine to a coarse grid. *International Journal for Numerical Methods in Engineering*, 12(1):35–45, 1978.
- [17] Feiteng Huang and Xiaoping Xie. A modified nonconforming 5-node quadrilateral transition finite element. *Adv. Appl. Math. Mech.*, 2(6):784–797, 2010.
- [18] Igor Kossaczky. A recursive approach to local mesh refinement in two and three dimensions. *Journal of Computational and Applied Mathematics*, 55(3):275–288, 1994.
- [19] Sai-Huen Lo, KH Wan, and Kam-Yim Sze. Adaptive refinement analysis using hybrid-stress transition elements. *Computers & structures*, 84(31):2212–2230, 2006.
- [20] JM McDill, JA Goldak, AS Oddy, and MJ Bibby. Isoparametric quadrilaterals and hexahedrons for mesh-grading algorithms. *Communications in applied numerical methods*, 3(2):155–163, 1987.
- [21] DJ Morton, JM Tyler, and JR Dorroh. A new 3D finite element for adaptive h-refinement in 1-irregular meshes. *International journal for numerical methods in engineering*, 38(23):3989–4008, 1995.
- [22] Theodore HH Pian. Derivation of element stiffness matrices by assumed stress distributions. *AIAA journal*, 2(7):1333–1336, 1964.
- [23] Theodore HH Pian. Some notes on the early history of hybrid stress finite element method. *International Journal for Numerical Methods in Engineering*, 47(1-3):419–425, 2000.
- [24] Theodore HH Pian and K Sumihara. Rational approach for assumed stress finite elements. *International Journal for Numerical Methods in Engineering*, 20(9):1685–1695, 1984.
- [25] Theodore HH Pian and Chang-Chun Wu. A rational approach for choosing stress terms for hybrid finite element formulations. *International Journal for Numerical Methods in Engineering*, 26(10):2331–2343, 1988.
- [26] Werner C Rheinboldt and Charles K Mesztenyi. On a data structure for adaptive finite element mesh refinements. *ACM Transactions on Mathematical Software*, 6(2):166–187, 1980.
- [27] Hanan Samet. The quadtree and related hierarchical data structures. *ACM Computing Surveys*, 16(2):187–260, 1984.
- [28] Alfred Schmidt and Kunibert G Siebert. *Design of adaptive finite element software: the finite element toolbox ALBERTA*, volume 42. Springer, 2005.
- [29] Robert Schneiders. Refining quadrilateral and hexahedral element meshes. *5th International Conference on Numerical Grid Generation in Computational Field Simulations*, pages 679–688, 1996.
- [30] Zhong-Ci Shi. A convergence condition for the quadrilateral Wilson element. *Numerische mathematik*, 44(3):349–361, 1984.
- [31] D Wu, Kam-Yim Sze, and Sai-Huen Lo. Two-and three-dimensional transition element families for adaptive refinement analysis of elasticity problems. *International journal for numerical methods in engineering*, 78(5):587–630, 2009.
- [32] Xiaoping Xie and Tianxiao Zhou. Optimization of stress modes by energy compatibility for 4-node hybrid quadrilaterals. *International journal for numerical methods in engineering*, 59(2):293–313, 2004.
- [33] Xiaoping Xie and Tianxiao Zhou. Accurate 4-node quadrilateral elements with a new version of energy-compatible stress mode. *Communications in Numerical Methods in Engineering*, 24(2):125–139, 2008.
- [34] Jinchao Xu. *Theory of multilevel methods*. Ph.D Thesis, Cornell University, 1989.

- [35] Guozhu Yu, Xiaoping Xie, and Carsten Carstensen. Uniform convergence and a posteriori error estimation for assumed stress hybrid finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 200(29):2421–2433, 2011.
- [36] Zhimin Zhang. Analysis of some quadrilateral nonconforming elements for incompressible elasticity. *SIAM journal on numerical analysis*, 34(2):640–663, 1997.
- [37] Xuying Zhao, Zhong-Ci Shi, and Qiang Du. Constraint-free adaptive FEMs on quadrilateral nonconforming meshes. *Journal of Scientific Computing*, pages 1–27, 2013.
- [38] Tianxiao Zhou and Yufeng Nie. Combined hybrid approach to finite element schemes of high performance. *International Journal for Numerical Methods in Engineering*, 51(2):181–202, 2001.